1

W. PAUL SCHUCK (SBN 203717)
pschuck@bztm.com

2

BARTKO ZANKEL BUNZEL & MILLER
A Professional Corporation

3

One Embarcadero Center, Suite 800
San Francisco, California 94111

4

Telephone: (415) 956-1900
Facsimile: (415) 956-1152

5

6

VICTOR G. HARDY (*pro hac vice*)
NICOLE E. GLAUSER (*pro hac vice*)

7

CHESTER J. SHIU (*pro hac vice*)
cshiu@dpelaw.com

8

DINOVO PRICE ELLWANGER & HARDY LLP
7000 North Mopac Expressway, Suite 350

9

Austin, Texas 78731
Telephone: (512) 539-2626

10

Facsimile: (512) 539-2627

11

Redacted

Attorneys for Plaintiff

12

SOFTWARE RIGHTS ARCHIVE, LLC

13

UNITED STATES DISTRICT COURT

14

NORTHERN DISTRICT OF CALIFORNIA

15

SAN JOSE DIVISION

16

17

18

SOFTWARE RIGHTS ARCHIVE, LLC,

19

Plaintiff,

20

v.

21

FACEBOOK, INC.,

22

Defendant.

Case No. 5:12-03970 RMW

**DECLARATION OF GREG BOSCH IN
SUPPORT OF MOTION FOR
ADMINISTRATIVE RELIEF PURSUANT TO
CIVIL L.R.7-11 FOR (1) LEAVE TO FILE
DECLARATION OF GREG BOSCH IN
SUPPORT OF DISCOVERY DISPUTE JOINT
REPORT #2; AND (2) AN ORDER FILING
THE DECLARATION OF GREG BOSCH
UNDER SEAL**

23

24

25

26

DOCUMENT SUBMITTED UNDER SEAL

27

AND CHAMBERS COPY

28

I, Greg Bosch, under penalty of perjury, hereby make the following declaration:

My name is Greg Bosch. I am over eighteen years of age and am fully competent to make this Declaration. I have personal knowledge of the facts herein and they are true and correct.

1.   I received a Bachelors Degree in Computer Science from Lehigh University, and was subsequently awarded a Masters of Science Degree from Lehigh.

2.   I have subsequently been employed as an independent software consultant since 2009 with respect to the patents in suit. I have extensive experience with the patents in suit and reviewing source code related to internet search companies. I have conducted patent infringement reviews of source code belonging to companies including Google, Yahoo, IAC (Ask.com), Cambridge Silicon Radio, Valve, and others.

3.   I was hired by Software Rights Archive, LLC (SRA) to study the functionality of Facebook's search systems, content display systems and sponsored link serving (*i.e.*, advertising) systems by reviewing relevant documents and conducting a review of Facebook's source code.

4.   In accordance with by duties, since January 14, 2013 I have reviewed the documentation produced by Facebook as part of its 3-4 obligations as well as the provided source code. Furthermore, in accordance with my duties, since February 5, 2013 I have reviewed voluminous amounts of code spanning thousands of files.

5.   At all times to my knowledge, Facebook has designated its source code at the highest level of protection under the Protective Order in this case, "HIGHLY CONFIDENTIAL – SOURCE CODE – OUTSIDE COUNSEL ONLY." Facebook has thus imposed strict limits on the time, location and manner of my review of their source code, which Facebook represents is not accessible publicly.

Review of Facebook's Technical Production

6.   Facebook's document includes about 2000 technical pages, produced from January 14, 2013 to March 27, 2013.

HIGHLY CONFIDENTIAL – SOURCE CODE

7.   Facebook's source code production has included more than 160+ million lines of code, including more than 79 million lines of code relevant to Facebook's operations as of the date of production, more than 37 million lines of code relevant to its operations as of January 2012, more than 26.5 million lines of code relevant to its operations as of January 2011, and greater than 16 million lines of code relevant to its operations as of January 2010.

8.   I have spent over 350 hours conducting a review of the source code produced by Facebook. Under my supervision, Bob O'Brien, another software consultant hired by Software Rights Archive has spent 650 hours conducting a review of Facebook's source code.   Under my supervision, William J. Laubenheimer, another software consultant hired by Software Rights Archive, has spent over 361 hours conducting a review of Facebook's source code. The total number of man hours spent reviewing Facebook's source code exceeds 1361 hours at a total cost of $180,000.

9.   I have also spent over 150 hours conducting a review of documents produced by Facebook and found from public sources in concert with my review of their source code.

10.   At the start of my review, Facebook had produced only 331 pages of documentation describing the functionality of the more than 160+ million lines of source code provided. As of May 31, 2013, it is my understanding that Facebook has produced only 2,335 pages of technical documentation (Bates Ranges 1–640 and 13191–14885). The overwhelming majority of these documents were very high level and not directed to the issues presented by these patents or were too ambiguous to be of use. Many of the documents were PowerPoint presentations and simply did not contain specific details about the subject matter that would have been of any use. There were also substantial amount of duplicate documents in the production. Given my experience with other technical productions, it would appear to me that Facebook has only produced a small fraction of its relevant documents. From the documents I have

seen, Facebook has a central repository of technical documents concerning its systems stored within a searchable wiki. It appears that only a small fraction of those documents describing the accused instrumentality have been produced. The wiki documents produced to date only cover very limited areas of the accused instrumentality and thus appear incomplete. Facebook's limited production of relevant documentation (despite its cache of wiki documents available to Facebook engineers), combined with its refusal to produce deponents for deposition or answers to SRA's interrogatories, has effectively withheld all guideposts to understanding the massive quantity of source code provided.

11. I have extensively reviewed Defendant's non-code technical production. For the most part, these documents are very high level and only cover a limited part of the accused instrumentality. These documents reveal that Defendant is in fact calculating values that analyze indirect relationships, however, with the exception of some cases, they do not identify the variables that correspond to these analyses so that SRA may locate them in the code. Nor do the technical documents state which search systems in fact use these analyses or even whether these analyses are used by search systems at all (or whether they are merely experimental).

12. SRA has accused five basic serving systems and numerous indexing and calculation systems. A given serving system alone is expressed by millions of lines of code. For example, the accused advertising/sponsored links serving system is ▮ million lines of code alone. These systems are extremely complex and consist of multiple subsystems. As another example, Facebook's ▮ system, which, based upon my review of Facebook's document and source code production, in my opinion corresponds to the accused instrumentalities News Feed and Timeline. This system includes at least three primary components including ▮ ▮. Among the source code for this Accused Instrumentality, ▮

different classes related to ██████████, over ██ classes related to ██ ██████ and █ classes of ██████ have been provided.

13. Unlike most source code reviews in patent cases, the systems relevant to the asserted claims run through the entirety of the source code base. There is no one folder or place to locate infringing functions. The source code base is so large that one cannot practically manually review the source code or even a single system. In order to review the source code, a researcher must rely on electronic searching to locate relevant functionality. Of course, to employ electronic searching one must have the name of the relevant variable or function.

14. Interrogatory No. 1 seeks to identify the necessary functions and corresponding variable names by requesting a listing of the numerical analyses considered or used by a search system. This, among other things, is necessary to enable electronic searching of the code. These analyses are usually expressed in terms of numerical values and other representations that are retrieved by the search engine. In the industry, the terms "signal," "feature," "input," "rank," "factor," and "score," are generally used to describe the analyses considered by search system. Facebook's documentation internally often uses the terms "features" and "scores" to describe these analyses.

SRA was Unable to Locate the Responsive Information

15. I attempted to ascertain the information sought by Interrogatory No. 1 by spending months reviewing source code and documents but still could not sufficiently locate the responsive information. I have compiled a list of analyses that I believe is relevant after the review to date and these analyses are identified in Appendix A to Discovery Dispute Joint Report #1: Interrogatory No. 1 (Subparts A, C, D and G) and correspondence referenced therein. I believe that this list is only a fraction of the relevant analyses that may use "link data" (*i.e.*, "signals," "features," "inputs," "ranks," "factors," and "scores") requested by Interrogatory No. 1 (subparts A and C).

HIGHLY CONFIDENTIAL – SOURCE CODE

After this review of the code, I have determined that it is not possible to identify all of the relevant features and corresponding variables by merely manually reviewing the code. Although the code may refer to many variable names, one would have to manually read most of the millions of lines of the code base to even identify all of the variable names used by the systems. Since variables are subsumed within other variables during the calculation of the feature and within the search systems that use features, one would have to read substantially all of the calculation code and all of the serving code (most of the 160 million lines of code) just to produce a list of variable names for the features. The manual reading of all this code is not possible on the three search terminals provided by Facebook within the period allowed for SRA to update its infringement contentions. I also note from my experience reviewing search engines in other cases that Facebook would certainly have design, testing, quality, and development documentation listing the features and scores that its systems use. Consistent with my understanding, FB-SRA-0000407 depicts a partial sample from such a list:

Redacted

1   Likewise, FB-SRA0015738 indicates that Facebook engineers have tools that allow a

2   Facebook engineer to print out, in human readable form, all the features used in a

3   model (*e.g.*, "███████████" see below):

4

5

6

7

8

9
                                  Redacted
10

11

12

13

14

15

16

17   Thus, Facebook can identify the basic feature set used by its systems using automated

18   tools without having to read the entirety of the source code.

19   16.   Even if one has all of the variable names, the names themselves do not reliably

20   indicate which features and scores use link data. For example, names like

21   "████████████████████" and "██████████████████████" are

22   names of features that leave one to guess at the nature of the analyses. *See* Facebook

23   Counsel's May 28, 2013 Email (identifying ███████████████, which includes the

24   aforementioned names). Other references may be simply the number "15" and that

25   reference is not maintained consistently throughout the code. Without knowing the

26   type of data being loaded in the accessed data structures for the calculation of each

27

28

feature, with the exception of certain cases, one cannot necessarily determine whether the feature uses link data by review of code alone.

17.   The identification of the system that uses a feature or other value sought by subpart D of Interrogatory No. 1 is critically important to enable SRA's infringement analysis of the code. Much of our source code review has been dedicated to trying to find out whether given search features are actually used for searching rather than for other functions. Indeed, a given features may be experimental. We have had particularly difficulty in matching a given feature to a given search system based upon the information to date. Typically, a search "feature" or other similar analysis requested by Interrogatory No. 1 is calculated in one place in the code and then used by one of the above serving systems in another place in the code.   Although the "feature," rank or other value may have one variable name after it is calculated, this will change a multitude of times before it is considered by the serving system. This is because the feature value is combined with other feature values to create composite ranks and subsumed into other feature values and ranks. Features and other relevant values will be saved under one particular name and used under a different variable name or id. By the time that the feature variable is used to serve results, it will have been subsumed into a multitude of different variable names. Thus, even if SRA was given all of the initial feature variable names, it cannot identify the systems that use them without additional information from Facebook and manually following the trail from start to finish through tens of millions of lines of code. Performing this task for all the features is practically impossible in the allotted time without the specialized tools utilized by a Facebook engineer and other guidance tools. Also, in the latter case where a variable name is simply saved under a different name or keyed to an id, it would be impossible to track the variable names without an identification of each instance this happens. In contrast, one would expect that a search engineer would know the basic system that uses a given search feature or have reference material

1   describing this. In addition, Facebook appears to have specialized systems not made

2   available to my team that track functionality, variables and scores as they are used

3   throughout their systems. Facebook document FB-SRA0015778–79, which provides

4   directions for automatically building "call graphs," a common term in the art referring

5   to a flowchart diagram that tracks the execution of code and its required inputs (*i.e.*,

6   features). A depiction of its output is set forth below:

7

8

9

10

11

12

13

14

15

16   Redacted

17

18

19

20

21

22

23

24

25

26

27

28

1
2
3
4
5
6
7
8
9
10
11
12

Redacted

13   From the instructions in the document, these systems can create a flowchart that
14   tracks the use of a given feature throughout its code and can match the feature with
15   the serving systems that use the feature. Thus, it appears that a Facebook Engineer
16   can use this software to easily determine the information requested in Interrogatory
17   No. 1 Subpart D without resorting to many months of source code review.
18   Furthermore, I believe the engineers who develop the systems that calculate scores
19   would certainly know the inputs that use link data requested by Interrogatory No. 1. I
20   believe a Facebook Engineer would not have to review all of the code to find such an
21   answer.

22   18.   Without knowing the basic information of what features apply to what systems sought
23   by Interrogatory No. 1 subpart D, we also find ourselves often trying to prove a
24   "negative." In those cases where a system does not use the variable, you are forced to
25   read the entire source code base to reach the conclusion that the variable is not being
26   used.   This is extremely inefficient. Thus, knowing the systems for which the
27   variable is used is critically important to any efficient review of the code.

28

HIGHLY CONFIDENTIAL – SOURCE CODE

Facebook Withheld the Most Crucial Parts of the Accused Instrumentality

19. One reason that SRA had difficulty in locating the systems that use some of the features identified by SRA was that it appears that Facebook withheld from its production the "models" of its machine learned systems. Machine learning is a branch of artificial intelligence that involves the study, design and construction of systems capable of learning from data. A machine learning algorithm, as its name suggests, is a framework for learning how to solve a problem given a set of existing data. Feature inputs are put into the systems and the system then analyzes the inputs and constructs a "model." In the case of the systems that create or use ranks, scores and features, the model defines how the inputted features are combined to calculate a score or other rank. It is the model that defines whether a given input feature is actually used and how it is used.
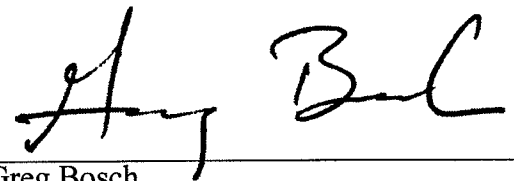
20. According to Facebook's document and source code production, Facebook uses machine learning algorithms pervasively in its systems, including the accused instrumentalities. Examples include, but are not necessarily limited to, ████████, People You May Know, News Feed, Timeline, and Search. Indeed, Search alone has at least ██ different machine learning components, including ███████████████ ██ ████████████████████████████.

21. We have been unable to locate the models for any of the above system. We have confirmed that the model for ██████████ is not located in the place where it is indicated to be (*i.e.*, ██████████, *see* file ██████████████, ln. 8 and 113). We believe that none of the models have been provided (at least in any human readable format). Without the models it is impossible to do a complete infringement review for all the systems or begin to determine whether any feature is actually used. The failure to provide these models

HIGHLY CONFIDENTIAL – SOURCE CODE

1   has resulted in a massive amount of wasted time searching the code base and other

2   source code materials.   We have spent **3 months** looking for the uses of identified

3   features by the serving systems through millions of lines of code.   It was impossible

4   for us to have located this use because it would have been indicated in the models not

5   provided.

6   22.   I have reviewed the data identified in Facebook counsel's May 28, 2013 letter and,

7   based upon my review of that later, my review of Facebook's document production,

8   and my review of Facebook's source code, in my opinion the identified files only

9   disclose a single set of inputs for a single learned algorithm, *e.g.*, ████████, and

10   not the learned model produced by applying the machine learning algorithm in

11   question.   I have not been provided any evidence that this set of inputs is complete or

12   used by Facebook in production, *i.e.*, to provide results to end users.   It certainly does

13   not address the vast majority of missing information for the accused systems.

14

15   I declare under penalty of perjury under the laws of the United States of America that the

16   foregoing is true and correct.

17

18

19   Greg Bosch
    Executed on June 7, 2013
20   Fairfield, CT

21

22

23

24

25

26

27

28